$2.60

55-3437 C.1

SOME APPLICATIONS OF THE THEORY OF DYNAMIC PROGRAMMING—A REVIEW

Richard Bellman

P-490

Revised 14 May 1954

The RAND Corporation
1700 MAIN ST. • SANTA MONICA • CALIFORNIA

is provided.

~~Summary: The purpose of this paper is to provide~~ an expository account of the theory of dynamic programming. To illustrate the general principles, two particular problems, one of deterministic type and one of stochastic type, are treated.

SOME APPLICATIONS OF THE THEORY OF DYNAMIC PROGRAMMING—A REVIEW

Richard Bellman

§1. Introduction

In this expository paper, dedicated to an introduction to and an illustration of the techniques of the theory of dynamic programming, we shall consider two problems of rather simple form. are considered;

Problem (1) (Optimal Allocation).

We are given a resource, x, to divide into two parts, y and x—y. From y we obtain a return of g(y); from x—y a return of h(x—y). In so doing, we expend a certain amount of the original quantity and are left with a new quantity, ay + b(x—y), where $0 < a, b < 1$. This process is now continued. The problem is to ~~How does one~~ allocate at each stage so as to maximize the total return obtained over a finite or unbounded number of stages.

Problem (2) (Efficient Gold Mining).

We are fortunate enough to possess two gold mines, Anaconda and Bonanza, the first of which contains an amount x of gold, while the second possesses an amount y. In addition, we have a rather delicate gold-mining machine which has the property that if used to mine gold in Anaconda, there is a probability p_A that it will mine a fraction r_A of the gold there and remain in

working order, and a probability $(1-p_A)$ that it will mine no gold and be damaged beyond repair. Similarly, Bonanza has associated the probabilities p_B and $(1-p_B)$ and the fraction r_B.

We begin by using the machine in either the Anaconda or Bonanza mine. If the machine is undamaged, we again make a choice of using the machine in either of the two mines, and continue in this way, making a choice before each mining operation, until the machine is damaged.

What sequence of choices maximizes the amount of gold mined before the machine is damaged?

Insofar as these problems involve multi-stage processes, large numbers of variables (when formulated in classical terms), chance events (in the second case), and the determination of policies rather than functions, they typify a very large set of important and difficult problems which have arisen in recent years to plague the economist, industrialist, strategist, and through them, the mathematician.

The methods we shall employ to treat the above questions constitute a part of the theory of dynamic programming, a mathematical theory which has been created over the last few years specifically to meet the challenge of these problems. Applications of the theory have already been made to the theory of investment and allocation, to logistics, to testing and learning theory, to problems of purchasing and inventory, to scheduling, to the planning of industrial and economic processes, and to control problems in engineering and economics.

§2. Optimal Allocation—Classical Formulation

Let us now see how Problem 1 above would be attacked, employing conventional techniques.

If there is only one stage to the process, the total [redacted] return is

$$(2.1) \qquad R_1(x,y) = g(y) + h(x-y).$$

The problem of maximizing $R_1(x,y)$ over y in $[0,x]$ is one which may be solved readily by means of calculus, or graphically.

If there are two stages, let y_1 be the choice in the first step and y_2 the choice at the second; then

$$(2.2) \qquad R_2(x,y_1,y_2) = g(y_1) + h(x_1-y_1) + g(y_2) + h(x_2-y_2),$$

where

$$(2.3) \qquad x_1 = x, \; x_2 = ay_1 + b(x_1-y_1),$$

and y_1 and y_2 are constrained by

$$(2.4) \qquad 0 \leq y_1 \leq x_1, \quad 0 \leq y_2 \leq x_2.$$

Quite generally, if there are N stages, the total return due to successive allocations of $y_1, y_2, \cdots, y_N$ will be

$$R_N(x_1,y_1,y_2,\cdots y_N) = g(y_1) + h(x_1-y_1) + g(y_2) + h(x_2-y_2) + \cdots g(y_N) + h(x_N-y_N), \tag{2.5}$$

where

$$\begin{aligned} x_1 &= x, \\ x_2 &= ay_1 + b(x_1-y_1) \\ &\vdots \\ x_N &= ay_{N-1} + b(x_{N-1} - y_{N-1}), \end{aligned} \tag{2.6}$$

and $(y_1,y_2,\cdots y_N)$ lies in the region

$$R: \quad \begin{aligned} 0 &\le y_1 \le x_1, \\ 0 &\le y_2 \le x_2 \\ &\vdots \\ 0 &\le y_N \le x_N \end{aligned} \tag{2.7}$$

Even for small n the problem of determining the maximum of R_N over the region described by the inequalities of (2.7) is a problem of formidable proportions, particularly since some of the extremum points may be at endpoints, thus rendering a direct application of calculus impossible.

§3. Optimal Allocation—Functional Equation Approach

The key to a different and more fruitful approach to Problem 1 is the petulant comment that the conventional approach provides too much information, far more than the practical man carrying out the process needs. He does not need the values of $y_1, y_2, \cdots$, and y_N; he needs only the value of y_1, given N and x.

Let us then use this observation to provide a different formulation.

To begin with, let us call any choice of $y_1, y_2, \cdots y_N$, for an N-stage process, a <u>policy</u>, and call any policy which yields maximum value of $R_N(x, y_1, y_2, \cdots, y_N)$ an <u>optimal policy</u>. Observing that the total return obtained using an optimal policy depends only upon x, the initial quantity of money, and N, the number of stages, we define

(3.1) $f_N(x)$ = total return obtained from an N-stage process given an initial amount x and employing an optimal policy.

Using this notation, let us compute the total return obtained using an initial division of x into y and x-y in the first step of an N-stage process. The immediate return due to the initial allocation will be $g(y) + h(x-y)$, and we will have $ay + b(x-y)$ with which to continue for N-1 remaining stages. It is clear that whatever the choice of y initially, the remaining amount, $ay + b(x-y)$, will be used optimally for the N-1 remaining stages, yielding, therefore, a further return of $f_{N-1}(ay + b(x-y))$. Hence, the total N-stage return due to an initial allocation of y will be

$$R_N(x,y) = g(y) + h(x-y) + f_{N-1}(ay + b(x-y)). \tag{3.2}$$

By definition,

$$\begin{aligned} f_N(x) &= \max_{0 \leq y \leq x} R_N(x,y) \\ &= \max_{0 \leq y \leq x} \Big[g(y) + h(x-y) + f_{N-1}(ay + b(x-y)) \Big]. \end{aligned} \tag{3.3}$$

This is the basic functional equation for the sequence $f_N(x)$. Its importance lies in the fact that it translates a problem in policy space into one in the more familiar function space.

§4. Computational Techniques

Let us now see what we have accomplished by converting the problem from that of maximizing the function of N variables in (2.5) to that of determining the sequence $\{f_N(x)\}$. In the first place, we have presented ourselves with a nonlinear sequence of functional equations possessing all the difficulties attendant upon nonlinear equations. In return, however, we have reduced the dimensions of the problem from N to 1 and thus considerably the analytic and computational aspects of the problem.

Beginning with $f_1(x)$, which is given by

$$f_1(x) = \max_{0 \leq y \leq x} \Big[g(y) + h(x-y) \Big], \tag{4.1}$$

we may compute $f_2(x)$, $f_3(x)$, and so on, using (3.3). In the

course of the computation of $f_N(x)$ we automatically compute $y(x) = y_N(x)$, which is actually the essential information.

Conversely, given $y_N(x)$ for each N and x we may compute $f_N(x)$ recursively. We have then a duality between the maximum return, $f_N(x)$, and the optimal policy, symbolized by $y_N(x)$. A knowledge of either enables the other to be computed.

Let us now exploit this fact. Since the amount remaining after each stage decreases geometrically, it is clear that for large N there will be little difference between $f_N(x)$ and $f_{N+1}(x)$, assuming, of course, that $g(0) = h(0) = 0$ and that g and h are continuous near zero. It follows that for large N we may write

$$(4.2) \qquad f(x) = f_{\infty}(x) \cong f_N(x)$$

and replace the sequence of equations in (3.3) by the one equation

$$(4.3) \qquad f(x) = \max_{0 \le y \le x} \left[g(y) + h(x-y) + f(ay + b(x-y)) \right].$$

This equation, with the solution fixed by the requirement that $f(0) = 0$, may now be solved by successive approximations. One set of approximations is, of course, the sequence $\{f_N(x)\}$ determined above. However, we may do much better in the following way: Instead of seeking approximations in function space, let us look for approximations in policy space; which is to say, instead of approximating to $f(x)$, the maximum return, let us approximate to $y(x)$, the optimal allocation.

In many of these problems, experience will have yielded a great deal of information concerning optimal policies, and it is precisely in this type of approximation that this experience can be put to best use.

Let us illustrate: In solving (4.3), we may consider the following possible policies, each of which have some intuitive basis

(4.4)

(a) At each stage let $y = 0$ or x depending upon whether $g(x)/(1-a)x > h(x)/(1-b)x$ or not

(b) Choose y so that

$$\frac{g(y)}{(1-a)y} = \frac{h(x-y)}{(1-b)(x-y)}$$

Let $f_o(x) = f_g(x)$ be the return calculated by recurrence, using one of the other of these policies. We may now compute successive approximations by means of the relation

$$f_{N+1}(x) = \max_{0 \leq y \leq x} \left[g(y) + h(x-y) + f_N(ay + b(x-y)) \right]. \tag{4.5}$$

The important point to emphasize is that we clearly have

$$f_o(x) \leq f_1(x) \leq f_2(x) \cdots . \tag{4.6}$$

Thus each approximation is automatically an improvement.

§5. Some Typical Results.

Let us now present some typical results which may be obtained concerning the nature of the solutions of this new class of

functional equations. These results are important since they yield first approximations to the solutions f of more complicated equations.

<u>Theorem</u> 1. <u>If</u> $g(x)$ <u>and</u> $h(x)$ <u>are both strictly convex functions of</u> x, <u>an optimal policy requires that</u> $y = 0$ <u>or</u> x.

The situation where g and h are both concave is more complicated.

<u>Theorem</u> 2. <u>Let</u>

(a) $g(0) = h(0) = 0$,

(b) $g'(x), h'(x) \geq 0$, <u>for</u> $x \geq 0$,

(c) $g''(x), h''(x) \leq 0$, <u>for</u> $x \geq 0$,

<u>and consider the sequence of equations</u>

$$f_1(x) = \operatorname*{Max}_{0 \leq y \leq x} \left[g(y) + h(x-y) \right]$$

$$f_{n+1}(x) = \operatorname*{Max}_{0 \leq y \leq x} \left[g(y) + h(x-y) + f_n(ay + b(x-y)) \right], \quad n=1,2,\cdots$$

<u>For each</u> n, <u>there is a unique</u> $y_n = y_n(x)$ <u>which yields the maximum</u>. <u>If</u> $b < a$, <u>we have</u> $y_1 \leq y_2 \leq y_3 \leq \cdots$, <u>and the reverse inequalities for</u> $b > a$. <u>In particular, if</u> $y_n(x) = x$, <u>for some</u> n, <u>in the case</u> $b \leq a$, <u>then</u> $y_m(x) = x$ <u>for</u> $m \geq n$.

This result is useful for approximation purposes since y_1, y_2 and even y_3 may be determined by hand computation quite quickly.

Even when g and h are convex and we know that $y = 0$ or x, it is not easy to determine which is the correct y-value. The following result is useful for approximation purposes:

<u>Theorem 3.</u> <u>The solution of</u>

$$(5.1) \qquad F(x) = \text{Max}\left[cx^d + F(ax),\ ex^f + F(bx)\right]$$

<u>is given by</u>

$$(5.2) \qquad \begin{aligned} y &= x \ \underline{\text{for}}\ 0 \le x \le x_0 \\ &= 0 \ \underline{\text{for}}\ x_0 < x. \end{aligned}$$

<u>where</u>

$$(5.3) \qquad x_0 = \left[(c/(1-a^d))/(e/(1-b^d))\right]^{1/(f-d)}$$

Another particular case where the solution may be obtained simply is that where g and h are quadratic in x.

Let us now indicate briefly how Theorem 3, and other results concerning the solution of particular equations, may be used to obtain approximate solutions. Given two functions, $g(x)$ and $h(x)$, we may obtain an approximate solution to equation (4.3), if we can obtain approximations to $g(x)$ and $h(x)$ by means of functions of the type cx^d and ex^f. Replacing x by e^y, we see that this is equivalent to approximating to $g(e^y)$ by ce^{dy}, or to $\log g(e^y)$ by $\log c + dy$. Consequently, to obtain our approximate expressions, we plot $\log g(e^y)$ and $\log h(e^y)$ qua functions of y, and look for straight-line fits of the form $a + by$. This may readily be done by inspection.

Having obtained these approximations to g and h, we use Theorem 3 to find the exact solution of the approximate equation. This solution has an associated policy which may be used as an approximate policy for the original problem. This approximate policy, in turn, yields an approximate solution, which we may iterate, as above, to obtain monotone convergence.

In Theorems 1, 2, and 3 discussed above, we have shown how various important properties of the optimal policy are consequences of certain simple properties of stage-by-stage payoff functions. In order to determine the precise influence of these properties upon the degree of complication of the solution, we computed the solution of a problem in which g and h exhibited the "diminishing return" property. We took

$$(5.4) \qquad g(x) = e^{-10/x}, \quad h(x) = e^{-15/x},$$

and $a = .8$, $b = .9$, and computed $f(x)$, the solution of (4.3), by means of successive approximations.

Below, we see the curves for $f_1(x)$, $f_2(x)$, and $f(x)$. They illustrate the slowness of successive approximations based on successive stages, and the necessity for using the approximate techniques mentioned above if one wishes rapid convergence.

The curve for $y(x)$ given in Figure 4 illustrates the extreme complexity that may be expected in an optimal policy if we introduce functions which have points of inflections. Since these functions occur quite frequently in applications, as manifestations of the law of diminishing returns mentioned above, again the importance of approximation techniques is made clear.

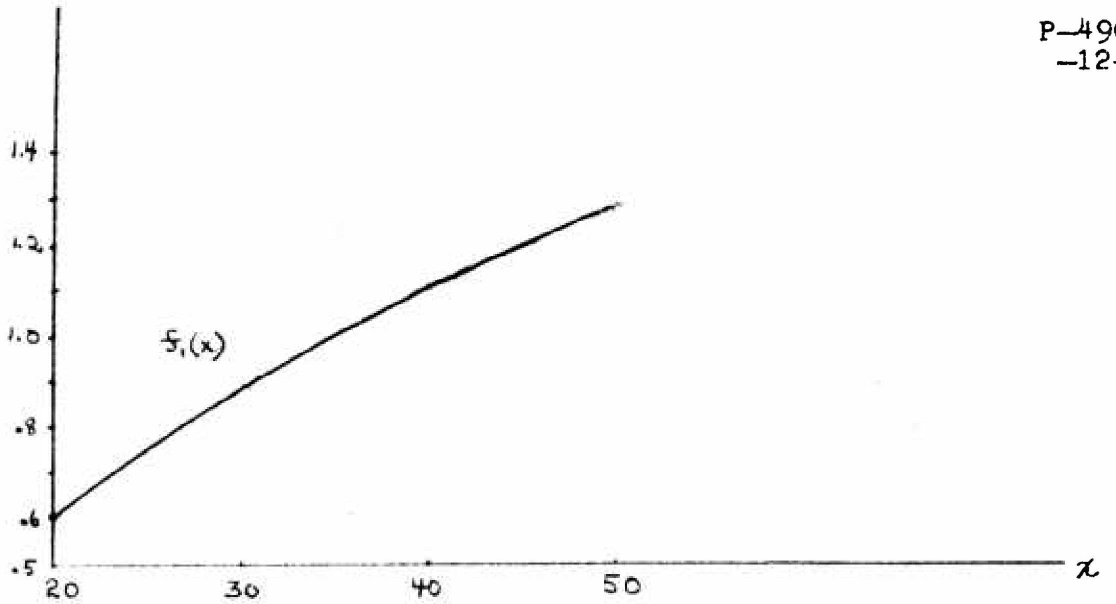


Fig. 1

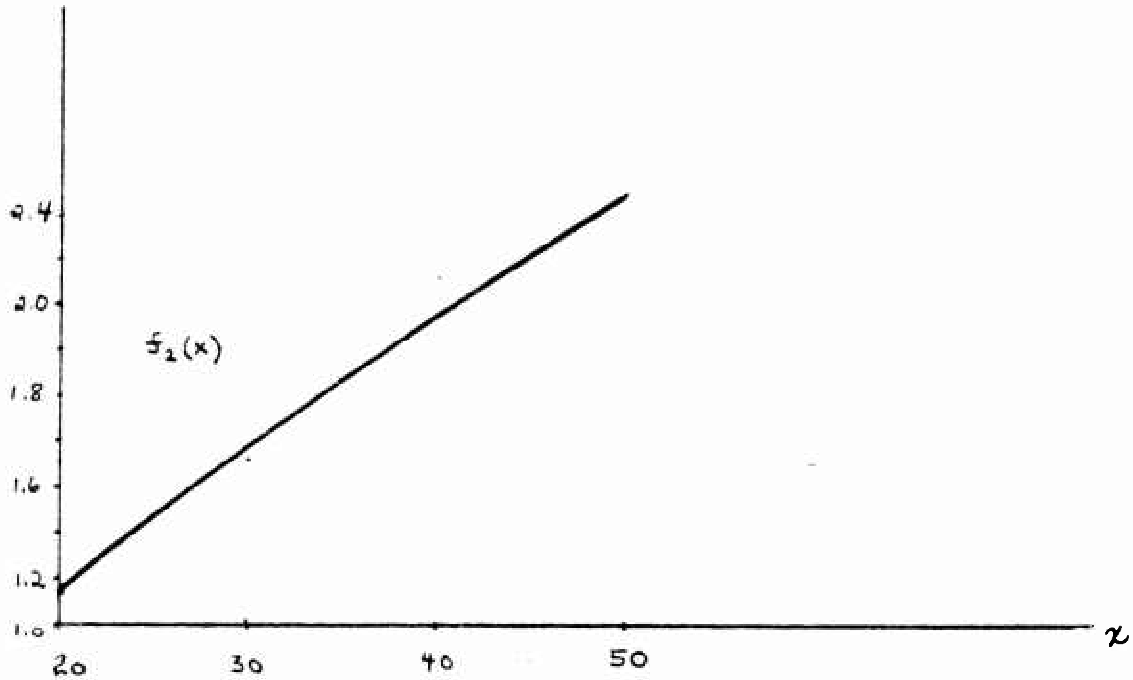


Fig. 2

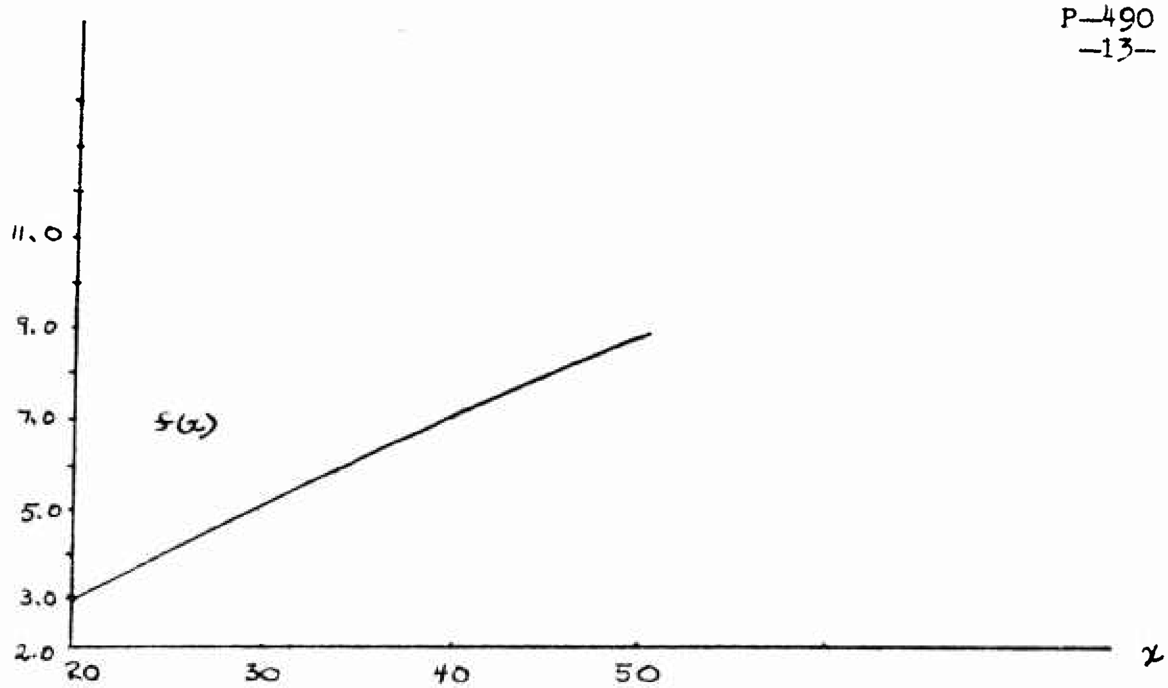


Fig. 3

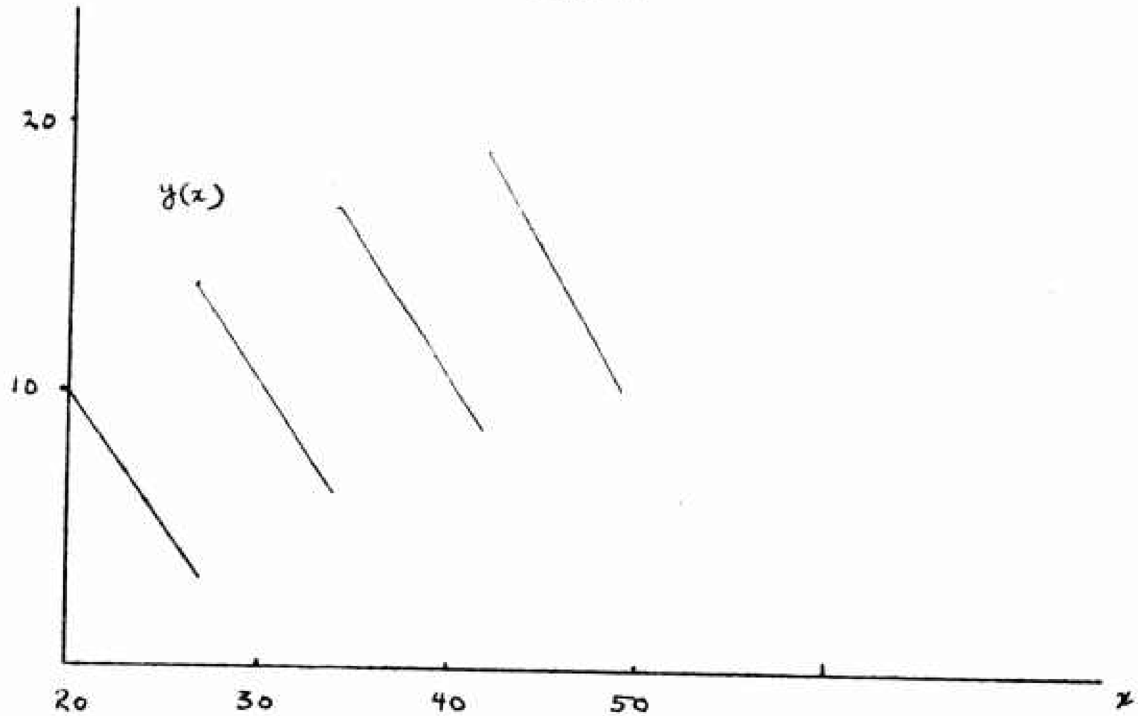


Fig. 4

§6. Gold Mining

Let us now consider the second problem, the one concerning the gold-mining machine of sensitive nature. This problem possesses an additional feature of difficulty due to presence of chance mechanisms.

A policy here will consist of a choice of A's and B's, which is to say, mining in Anaconda or in Bonanza. However, any such sequence such as

$$(6.1) \qquad S = AABBBABB\cdots,$$

must be read: A first, then A again if the machine is undamaged; then B is the machine still undamaged, and so on.

If initially, to avoid any conceptual difficulties inherent in unbounded sequences, we consider only mining processes which end automatically after N steps, regardless of whether the machine is damaged or not, it is quite easy to list all the possible policies.

Since we are dealing with a stochastic process, it is not possible to talk about the return from a policy.* We must console ourselves with some average of the possible returns. The simplest such is the usual average, or expected value.

Let us then agree that we are interested in the policy which maximizes the expected value of the amount of gold mined before the machine is damaged. Corresponding to every policy such as

---

* We might note in passing that this idea is a very difficult one to explain to a neophyte at card games, particularly in explaining the theory of a finesse.

(6.1), there will be an expected return. To determine an optimal policy it is merely necessary to list all possible policies, compute the expected returns and compare. Even if feasible, this method is clumsy and completely unrevealing as to the structure of an optimal policy.

§7. Functional Equation Approach

In place of the above enumerative approach, let us employ the functional equation technique of §3. Let us also simplify matters by going directly to the unbounded process. We define

(7.1) $f(x,y)$ = expected amount of gold mined before the machine is damaged when A has x, B has y, and an optimal policy is employed.

Let us compute the expected amount of gold mined if an A operation is used first, a quantity we denote by $f_a(x,y)$. The total expected amount will be $p_1 r_1 x$, as a result of the initial stage, plus the expected amount mined from the second stage on. It is clear that an optimal policy will be pursued from this point on if the machine survives. Hence, the expected amount obtained from the second stage on will be $f((1-r_1)x,y)$, since Anadonda now possesses $(1-r_1)x$ and Bonanza still has y. Thus,

$$(7.2) \qquad f_a(x,y) = p_1 \left[r_1 x + f((1-r_1)x,y) \right].$$

Similarly,

$$f_b(x,y) = p_2 \left[r_2 x + f(x,(1-r_2)y) \right]. \tag{7.3}$$

Since we wish to choose A or B so as to maximize the over-all expected return, we have

$$f(x,y) = \text{Max} \left[f_a(x,y),\ f_b(x,y) \right], \tag{7.4}$$

which yields the functional equation,

$$f(x,y) = \text{Max} \left\{ \begin{array}{ll} A: & p_1 \left[r_1 x + f((1-r_1)x,y) \right] \\ B: & p_2 \left[r_2 y + f(x,(1-r_2)y) \right] \end{array} \right\} \tag{7.5}$$

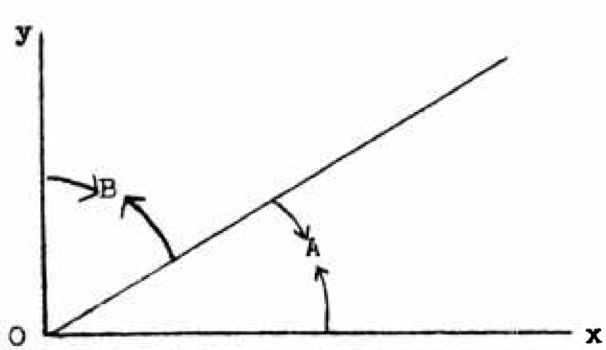
§8. <u>The Solution</u>

It may be shown, cf. [2], [3], [17], that the solution to (7.5) is given by

$$\begin{array}{lll} \text{(a)} & \text{if } \dfrac{p_1 r_1 x}{1-p_1} > \dfrac{p_2 r_2 y}{1-p_2}, & \text{take the A choice} \\ \text{(b)} & \text{if } \dfrac{p_1 r_1 x}{1-p_1} < \dfrac{p_2 r_2 y}{1-p_2}, & \text{take the B choice} \\ \text{(c)} & \text{if } \dfrac{p_1 r_1 x}{1-p_1} = \dfrac{p_2 r_2 y}{1-p_2}, & \text{either choice is optimal} \end{array} \tag{8.1}$$

Geometrically,

(8.2)



$$\frac{p_1 r_1 x}{(1-p_1)} = \frac{p_2 r_2 y}{(1-p_2)}$$

Observe that this type of solution is ideally suited to a problem involving chance effects. It tells what to do next in terms of where one is. Clearly, if from every position, the next move is determined, one can determine all possible optimal sequences. However, in this case as in so many similar cases, the solution is most clearly presented in the above form.

For further details concerning problems of this type, we refer to [2], [3], [9], [16], and [17].

§9. Discussion of the Solution

One of the principal reasons for attacking problems of the above type, which are extremely idealized and simplified versions of problems occurring in applications, lies in the fondly cherished hope that the pattern of the solution may make itself clear. Interpreting the mathematical solution in terms of intuitive concepts, we may discover some metaphysical concept such as a "principle of least action" which we can apply to problems of more complicated type.

Let us see what interpretation we can give to the solution given in (8.1). The expression $p_1 r_1 x/(1-p_1)$ has as its numerator $p_1 r_1 x$, the immediate expected gain from an operation, while its denominator is $(1-p_1)$, the probability that the machine will be destroyed, which is to say, the immediate expected loss. The expression $p_2 r_2 y/(1-p_2)$ consists of a similar ratio.

Consequently, both expressions are ratios of immediate expected gain to immediate expected loss, and the optimal policy is to choose at each stage the operation which maximizes the ratio.

Although this policy is not an optimal policy for all such problems, it is an excellent rule-of-thumb, and one which may readily be applied.

§10. A General Description of Dynamic Programming Problems

Having given some simple examples of dynamic programming problems, let us now see if we can, in some general way, characterize these problems. They possess the following common features:

(10.1)

(a) Multi-stage processes are involved.

(b) At each stage, the state of the process is described by a small number of parameters.

(c) The effect of a decision at any stage is to transform this set of parameters into a similar set.

We have purposely left the description a bit vague, since we feel that it is the spirit of the problem rather than the letter which is significant. A certain amount of ingenuity is always required in attacking new questions, and no amount of axiomatics and rigid prescriptions can ever banish it.

Add to the above the following simple

Principle of Optimality: An optimal policy has the property that whatever the initial state and initial decision may be, the remaining decisions must constitute an optimal policy with regard to the state resulting from the first decision, and we have the basic ingredients of the theory of dynamic programming. The rest is mathematics and experience.

§11. Some Typical Problems

To illustrate the way problems of multi-stage type occur in these fields, let us cite some typical problems:

1. A Scheduling Problem: Suppose we have a number of different objects which must be processed by a number of machines of different type. We assume that each machine can process only one item at a time and that the machines must be used in a fixed order. Given the times required for each machine to process each item, in general different, in what order should the objects be processed so as to minimize the total time required to process the complete set of items?

2. A Logistics Problem: Over a period of years, it is necessary to purchase a number of different types of equipment with different job performance ratings, different costs, and different salvage or resale values, in order to perform a number of assigned tasks. How should money be allocated to purchase the different classes of equipment so as to minimize the amount of money required to do a certain job, or conversely, so as to maximize the job done for a given appropriation of money?

3. A Smoothing Problem: There is a fluctuating demand for a product which requires a certain production force of employees at any given time. If the actual number of employees is greater than required, a certain loss is incurred due to nonproductivity. On the other hand, a certain loss is incurred whenever new employees are hired. What production force should be maintained so as to minimize the total loss over some fixed time period?

4. <u>An Optimal Inventory Problem</u>: At some initial time we have a quantity of merchandise in stock and are given the information that at the end of one time period we will be required to deliver a certain quantity of this merchandise. The precise amount required is not known, but a distribution curve for the demand is known. To meet this demand we may order more merchandise at a cost depending upon the amount ordered. ~~If the demand exceeds the amount in stock, a penalty depending upon the amount ordered.~~ If the demand exceeds the amount in stock, a penalty depending upon the deficit is levied and the request is fulfilled as far as possible.

Assuming that the situation repeats itself periodically and that future costs are discounted at a fixed rate, what ordering policy minimizes the over-all expected cost?

5. <u>A Control Problem</u>: We are given an engineering system which is ruled by a system of differential or difference equations. To maintain the system in its desired state, it is necessary to exert some control, the mathematical manifestation of which is a forcing term.

It is desired to control the system in such a way that the total cost, which is compounded of the cost of deviation from the desired state, plus the cost of control, is a minimum.

6. <u>Economic Investment</u>: In managing a business enterprise, we have our choice of taking money out as immediate profit, or of reinvesting the money to enlarge the business and increase future profit. What reinvestment policy maximizes the total profit derived over a given time period?

7. Bottleneck Problems: Suppose that we have a complex of industries, as for example, steel, tool, and auto, all employed in the production of one particular item, such as autos. At any particular time we have our choice of allocating resources such as money, steel, and tools, to produce steel, tools, or autos, or to build steel factories, tool factories, or auto factories.

What allocation policy maximizes the total number of autos produced over a given time period?

8. Learning Theory: Suppose that we have two hundred critically ill patients and two new wonder drugs as yet untested. How should these drugs be tested on the patients so as to maximize the expected number of patients who are cured?

9. Testing Theory: Suppose we are testing a group of objects for a specific property and are given the probability, for each object, that the test will disclose this property if it exists, and the prior probability that each object has this property. What testing procedure will minimize the expected time required to determine a given number of objects with the required property?

For those interested in the mathematical treatment of these problems, we cite the following references:

Ad 1: 19, 14	Ad 4: 1, 18, 7	Ad 7: 10
Ad 2: 3, 4, 5, 8	Ad 5: 12	Ad 8: 20, 21, 22
Ad 3: 13	Ad 6: 12	Ad 9: 3

REFERENCES

1. Arrow, K. J., T. E. Harris, and J. Marschak, "Optimal Inventory Policy," Cowles Commission Paper No. 44, 1951.

2. Bellman, R., "On the Theory of Dynamic Programming," Proc. Nat. Acad. Sci., 38 (1952), pp. 716-719.

3. ————, An Introduction to the Theory of Dynamic Programming, RAND Report No. R-245, 1953.

4. ————, "On Some Applications of the Theory of Dynamic Programming to Logistics," RAND Paper No. P-457, 1953.

5. ————, "Some Problems in the Theory of Dynamic Programming," Econometrica, 22 (January 1954), pp. 37-48.

6. ————, "Dynamic Programming and a New Formalism in the Calculus of Variations," RAND Paper No. P-454, 1953.

7. ————, "On a Functional Equation Arising in the Problem of Optimal Inventory," RAND Paper No. P-480, 1954.

8. ————, "Computational Problems in the Theory of Dynamic Programming," RAND Paper No. P-423, presented at Symposium on Numerical Techniques, August 1953, Santa Monica, California.

9. ————, "Some Functional Equations in the Theory of Dynamic Programming," Proc. Nat. Acad. Sci., 39 (1953), pp. 1077-1082.

10. ————, "Bottleneck Problems and Dynamic Programming," Proc. Nat. Acad. Sci., 39 (1953), pp. 947-51.

11. Bellman, R. and D. Blackwell, "Some Two-person Games Involving Bluffing," Proc. Nat. Acad. Sci., 35 (1949), pp. 600-5.

12. Bellman, R., I. Glicksberg, and O. Gross, "On Some Variational Problems Occurring in the Theory of Dynamic Programming," Proc. Nat. Acad. Sci., 39 (1953), pp. 298-301.

13. ————————, "On Some Problems in the Theory of Dynamic Programming—A Smoothing Problem," (to appear).

14. Bellman, R., and O. Gross, "Some Combinatorial Problems Arising in the Theory of Multi-stage Processes," RAND Paper No. P-456, 1953.

15. Bellman, R., T. E. Harris, and H. N. Shapiro, "Studies in Functional Equations Occurring in Decision Processes," RAND Paper No. 382, 1952.

16. Bellman, R., and S. Lehman, "On the Continuous Gold-mining Equation," Proc. Nat. Acad. Sci., 40 (Feb. 1954), p. 115.

17. ————————————, "A Functional Equation in the Theory of Dynamic Programming and its Generalizations," RAND Paper P-433, 1953.

18. Dvortesky, A. J., J. Kiefer, and J. Wolfowitz, "The Inventory Problem—I: Case of Known Distributions of Demand," and "The Inventory Problem—II: Case of Unknown Distributions of Demand," Econometrica, 20, (1952), pp. 187-222.

19. Johnson, S., "Optimal Two- and Three-stage Production Schedules with Setup Times Included," RAND Paper No. P-402, 1953.

20. Johnson, S., and S. Karlin, "A Bayes Model in Sequential Design," RAND Paper No. P-328, 1952.

21. Thompson, W. R., "On the Likelihood that One Unknown Probability Exceeds Another in View of the Evidence of Two Samples," Biometrica, 25 (1953).

22. ——————————, "On the Theory of Apportionment," Amer. Jour. Math., 57 (April 1935).

bjc

AFBMD Library Charge-Out Card

Document No: 55-2437 Copy No: 1 O/A Report No: P 490

Title: SOME APPLICATIONS OF THE THEORY OF DYNAMIC PROGRAMMING A REVIEW

Class: (U)

Orig. Ag'y: RAND

Author: RICHARD BELLMAN Doc. Date: 14 May 54 Contract No:

Borrower's Name	Date Out	Date In	Borrower's Signature
PATTI LEE, STL	9/22/59		Patti Lee 1m